

智能简历解析系统建设
概要设计说明书
V1.0

猫猫狗狗队

文档修订记录

编号	版本号	修订时间	修订内容	修订人	审核人
1.	V1.0	2023.7.10	初始创建	蔡博凯	谢欣欧

目录

- 1 引言4
 - 1.1 编写目的.....4
 - 1.2 预期读者和阅读建议.....4
- 2 设计概述.....4
 - 2.2 设计原则和设计要求.....4
- 3 系统架构设计.....5
- 4 系统逻辑设计.....6
 - 4.1 系统组织设计6
 - 4.2 系统结构设计.....6
 - 4.3 功能模块设计.....8
 - 4.3.1 文件上传模块.....9
 - 4.3.2 简历解析模块.....9
 - 4.3.3 人岗匹配模块.....10
 - 4.3.4 信息展示模块.....10
 - 4.4 数据流和处理逻辑设计.....10
 - 4.5 系统接口设计.....11
 - 4.6 系统出错处理设计.....12
- 5 技术设计.....12
 - 5.1 前端详细技术.....12
 - 5.2 后端详细技术.....12
 - 5.3 数据库详细技术.....12
 - 5.4 测试详细技术.....12

- 表格 1 系统组织设计.....6
- 表格 2 服务器端系统特性表.....7
- 表格 3 管理端系统特性表.....7
- 表格 4 用户端系统特性表.....7
- 表格 5 功能模块系统特性汇总.....8
- 表格 6 文件上传系统特性详细描述.....9

- 图表 1 系统架构图.....6
- 图表 2 系统特性结构图.....8
- 图表 3 数据流和处理逻辑设计图.....11

1 引言

1.1 编写目的

本概要设计文档旨在介绍智能简历解析系统的高层设计，为读者提供对系统整体架构和设计原则的理解。本系统的目标是利用人工智能技术，提供一个高效、智能的简历管理和分析平台，帮助招聘人员和求职者更好地匹配岗位需求和人才需求。

1.2 预期读者和阅读建议

本文档可能的读者及阅读建议：

- ❖ 开发人员：应全篇认真阅读
- ❖ 项目经理：重点阅读设计概述
- ❖ 测试人员：应全篇认真阅读
- ❖ 文档编写人员：重点阅读设计概述

2 设计概述

本节描述现有开发条件和需要实现的目标，说明进行概要设计时应该遵循的设计原则和必须采用的设计方法。

2.1 限制和约束

系统开发的限制与约束条件为：

- ❖ 开发环境：Vs Code、IntelliJ IDEA、Navicat、Postman
- ❖ 时间限制：2023 年 7 月 15 日前
- ❖ 资源限制：需要使用移动云的第三方 API

2.2 设计原则和设计要求

软件系统进行概要设计的原则有：

❖ 命名规则：系统模块、变量等命名按照功能的英文名命名，并且采用驼峰命名法。

❖ 模块化设计原则：系统应采用模块化的设计方法，将功能划分为独立的模块，以便于维护和扩展。模块之间应该具有松耦合的关系，以便于单独开发和测试。

❖ 可扩展性原则：系统设计应具备可扩展性，能够支持未来的功能扩展和变化需求。通过松耦合的架构和可插拔的组件设计，以便于添加新功能模块和集成第三方组件。

❖ 系统易操作性要求：系统输入尽可能采用“选择”而非直接输入。

❖ 系统可维护性要求：系统设计应注重可维护性，包括代码的可读性、可理解性和可修改性。使用清晰的命名和结构化的代码组织，遵循良好的编码规范，以便于团队成员理解和维护代码。

❖ 性能优化原则：系统设计应考虑性能优化的方法，以提高系统的响应速度和吞吐量。合理选择数据结构和算法，减少资源占用和提高系统效率。

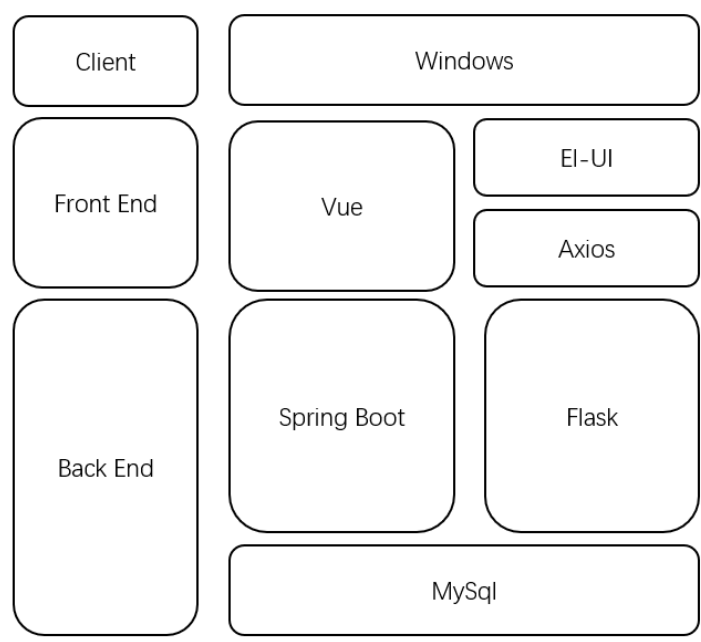
❖ 用户体验原则：系统设计应注重用户体验，提供直观、易用的界面和操作流程，以方便用户上传、管理和分析简历。

❖ 数据完整性和一致性原则：系统设计应保证数据的完整性和一致性。采用合适的数据库设计和事务管理，确保数据的正确性和一致性。

3 系统架构设计

智能简历解析系统采用了 B/S 的实验操作流程，根据 MVVM 的设计思想，对系统做出如下划分：视图层、控制层、业务逻辑层、数据持久层，使用了主流的后端框架 SpringBoot、使用 Python 编写的轻量级 Web 应用框架 Flask 与主流的前端框架 Vue，在系统界面使用了 EL-UI 模组，使用 Axios 模组进行前后端之间的信息传输，力求让用户获得良好的使用体验。

整个系统架构如图表 1 所示



图表 1 系统架构图

4 系统逻辑设计

4.1 系统组织设计

表格 1 系统组织设计

子系统编号	英文名称	中文名称	业务职能	安装地点	备注
1	Service	服务器端	数据的储存和处理	服务器端	
2	Administer	管理端	维护系统、录入简历数据	企业终端	
3	User	用户端	进行系统的使用	用户终端	

4.2 系统结构设计

本节将对系统特性（用户需求、系统功能）作较为详细的描述，并给出系统特性结构图。

表格 2 服务器端系统特性表

子系统编号：1 英文名称：Service 中文名称：服务器端				
特性编号	英文名称	中文名称	操作功能	备注
Service_1	Store	储存	数据的储存	
Service_2	Deal	处理	数据的调用、处理	
说明：				

表格 3 管理端系统特性表

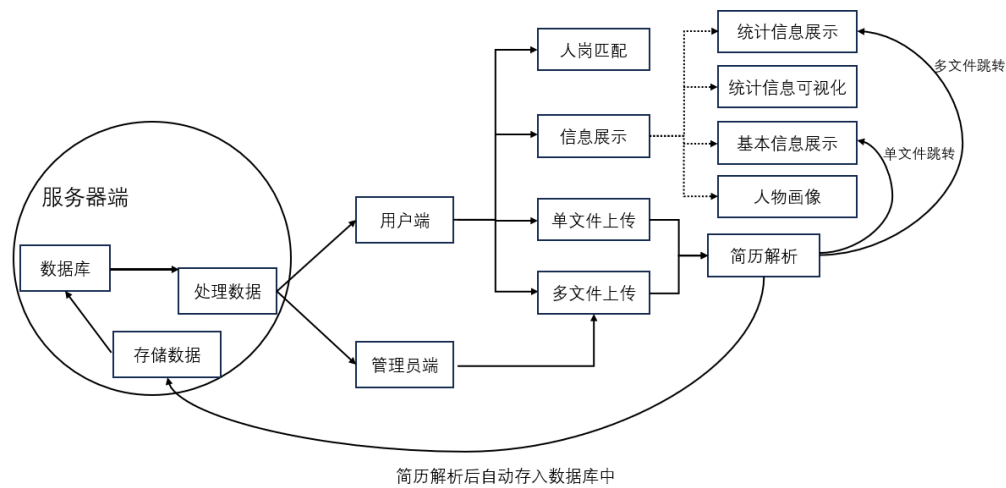
子系统编号：2 英文名称：Administer 中文名称：管理端				
特性编号	英文名称	中文名称	操作功能	备注
Administer_1	Init	录入数据	录入简历数据	
说明：管理员需要在一开始使用多文件上传功能录入简历，因为数百份简历的解析与上传需要十几分钟，此举可以减少用户的等待时间				

表格 4 用户端系统特性表

子系统编号：3 英文名称：User 中文名称：用户端				
特性编号	英文名称	中文名称	操作功能	备注
User_1	SingleLoad	单文件上传	上传一份简历	
User_2	MultiLoad	多文件上传	上传多份简历	
User_3	JobMatch	人岗匹配	根据 HR 提供的目录和数据库里的简历信息进行匹配度分析	
User_4	MultiData	统计数据展示	展示统计信息	
User_5	MultiAna	统计数据可视化	分析统计信息	

User_6	SingleData	基本信息展示	展示基本信息	
User_7	SingleAna	人物画像	分析基本信息	
说明：				

系统特性结构图



图表 2 系统特性结构图

4.3 功能模块设计

功能模块的系统特性表如表格 5 所示。

表格 5 功能模块系统特性汇总

功能模块		优先级	版本	状态	需求规格文档参考
文 件 上 传	单文件上传	高	1.0	已完成	表格 13、图表 7
	多文件上传	高	1.0	已完成	表格 13、图表 8
简历解析		中	1.0	已完成	表格 14
人岗匹配		高	1.0	已完成	表格 15、图表 11
信 息 展 示	基本信息展示	中	1.0	已完成	表格 16、图表 10
	人物画像	中	1.0	已完成	表格 17、图表 9
	统计信息显示	中	1.0	已完成	表格 18、图表 12
	统计信息可视化	中	1.0	已完成	表格 19、图表 13

4.3.1 文件上传模块

文件上传模块的系统特性详细描述见表格 6。

表格 6 文件上传系统特性详细描述

名 称	文件上传	标 识	Load
接 口 说 明	输入	上传的简历文件	
	输出	上传文件的存储路径	
处理逻辑	在客户端，用户可以通过拖放、多选或其他方式选择单个或多个文件。这些文件被分割成数据块，并通过 HTTP 或其他协议一并发送给服务器。服务器端接收文件数据块，将其重新组合成各自的完整文件，并进行处理和存储。		
系统特性	<ul style="list-style-type: none">- 支持选择和上传单个或多个文件- 显示上传进度- 批量上传和处理多个文件- 错误处理和反馈		

4.3.2 简历解析模块

简历解析模块的系统特性详细描述见表格 7

表格 7 简历解析模块特性详细描述

名 称	简历解析	标 识	FileDeal
接 口 说 明	输入	上传文件的存储路径	
	输出	解析后的简历信息	
处理逻辑	使用自然语言处理和文本分析技术对简历进行解析，提取 关键信息		
系统特性	<div>- 解析简历文件，提取关键信息</div> <div>- 识别和提取姓名、年龄、学历、毕业院校、工作经验等</div> <div>- 高准确性和可靠性</div> <div>- 错误处理和容错机制</div>		

4.3.3 人岗匹配模块

人岗匹配模块的系统特性详细描述见表格 8

表格 8 人岗匹配模块特性详细描述

名 称	人岗匹配	标 识	JobMatch
接 口 说 明	输入	解析后的简历信息、招聘岗位要求	
	输出	匹配度排名前 20 的简历	
处理逻辑	通过对比岗位要求和简历信息的关键词、技能、经验等， 计算匹配度评分。再通过匹配度评分进行排名。		
系统特性	<div>- 根据招聘岗位要求和候选人简历进行匹配</div> <div>- 分析技能、经验、教育背景等方面的匹配程度</div> <div>- 提供匹配度评分和推荐结果</div>		

4.3.4 信息展示模块

信息展示模块的系统特性详细描述见表格 9

表格 9 信息展示模块特性详细描述

名 称	信息展示	标 识	Data
接 口 说 明	输入	解析后的简历信息或数据库里信息	
	输出	基本信息、人物画像、统计信息	
处理逻辑	将解析后的简历信息、统计信息和人物画像描述进行展示和呈现。或者直接访问数据库中已有的信息进行展示		
系统特性	<ul style="list-style-type: none">- 显示候选人的姓名、教育背景、工作经历等基本信息- 基于候选人的个人信息和行为数据生成人物画像- 统计候选人的数量、分类、学历、经验等信息- 将统计信息以图表、图形等形式可视化展示		

4.4 数据流和处理逻辑设计

系统的数据流和处理逻辑设计框图如下图表 3 所示。

人岗匹配	岗位信息 简历信息	匹配度高的 简历	前端输入岗位要求，后端自动遍历数据库中的简历并进行匹配
------	--------------	-------------	-----------------------------

4.6 系统出错处理设计

定义一套错误分类和错误代码，用于标识不同类型的错误和异常情况。

5 技术设计

5.1 前端详细技术

- 采用了 Webpack 的模块打包机制
- 采用了基于 vue 构建用户界面的渐进式框架，并采用了 vue 全家桶（vue-router 管理路由，vuex 管理状态，vue-cli 脚手架，axios 管理网络请求）。
- 采用了基于 vue 的成熟的组件库 Element UI 构建页面。

5.2 后端详细技术

采用了 Spring Boot 框架和 Flask 框架开发后端

5.3 数据库详细技术

采用了 MySQL 存储数据，并使用 Navicat 软件简化开发

5.4 测试详细技术

前期采用了 Postman 软件测试接口，后期直接通过浏览器测试